(12) UK Patent Application (19) GB (11) 2 323 455 (13) A

(43) Date of A Publication 23.09.1998

(21) Application No 9811313.7

(22) Date of Filing 27.05.1998

(71) Applicant(s)
Trafford Limited
(Incorporated in the United Kingdom)
66 Newland Street, WITHAM, Essex, CM8 1AH, United Kingdom

(72) Inventor(s)
Andrew Stephen Fiore

(74) Agent and/or Address for Service
Sanderson & Co
34 East Stockwell Street, COLCHESTER, Essex, CO1 1ST, United Kingdom

(51) INT CL[6]
G04G 1/00 , G06F 1/14

(52) UK CL (Edition P)
G3T TAAF THA THB
G4A AFT

(56) Documents Cited
US 4706189 A

(58) Field of Search
UK CL (Edition P) G3T TAAF THA THB
INT CL[6] G04G 1/00 , G06F 1/00 1/14
Online:WPI

(54) Abstract Title
Time correcting computer dongle

(57) A dongle for a computer has a real time clock source and a power supply for the source. The dongle connects to the printer port of a PC and on system start-up, a dongle control program is called and reads the time and date from the dongle real time clock source, and uses this information to set the PC's hardware real time clock and the operating system software clock. In this way, year 2000 compliance may be assured, for a PC which, by itself, is not fully compliant. Alternatively or additionally, a PC having an inaccurate or failed hardware real time clock may be caused to operate correctly.

GB 2 323 455 A

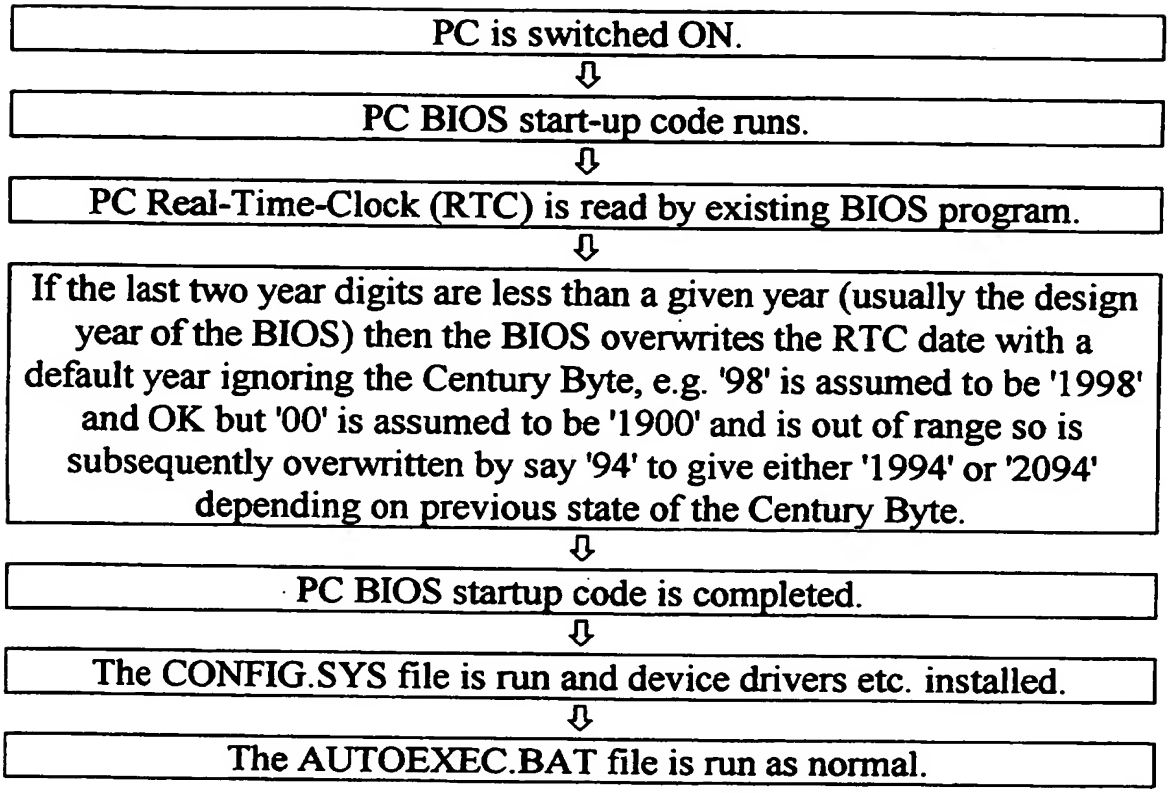
PC is switched ON.
PC BIOS start-up code runs.
PC Real-Time-Clock (RTC) is read by existing BIOS program.
If the last two year digits are less than a given year (usually the design year of the BIOS) then the BIOS overwrites the RTC date with a default year ignoring the Century Byte, e.g. '98' is assumed to be '1998' and OK but '00' is assumed to be '1900' and is out of range so is subsequently overwritten by say '94' to give either '1994' or '2094' depending on previous state of the Century Byte.
PC BIOS startup code is completed.
The CONFIG.SYS file is run and device drivers etc. installed.
The AUTOEXEC.BAT file is run as normal.

Figure 1

The first line of the AUTOEXEC.BAT file calls MILLWIZ.BAT which runs MILLTIME.EXE and ENFORCER.EXE reporting any loading problems.

⇩

MILLTIME.EXE checks to see if a RTC dongle exists

⇩

If dongle is present, then MILLTIME.EXE range checks the time and date of the dongle's RTC plus the state of the on-board rechargeable battery. If these are OK then MILLTIME.EXE reads the correct time and date from the dongle's RTC source, otherwise it prompts the user to enter the correct time and date which is used to set the dongle's RTC source.

⇩

The correct time and date are then written to the PC's RTC and software clock correcting both the power-off year 2000 rollover problem and any power-off leap day problem.

⇩

ENFORCER.EXE then loads a memory resident routine which intercepts any read or write access to the PC's RTC.

⇩

The remainder of the AUTOEXEC.BAT file is completed running any application files and handing over control to the operating system e.g. DOS or Windows and application programmes.

⇩

From then onwards, while the PC is still switched ON, the memory resident routine of ENFORCER.EXE intercepts ALL reads and writes to the PC's RTC setting the Century Byte to 19 for any year ending greater than or equal to 98 and to 20 for any year ending less than 98 before passing on the corrected date to the original calling program. Both the power-on year 2000 rollover problem and any power-on leap day problem are therefore resolved.

⇩

MILLTIME.EXE may be called again at any time with a '/t' appended to set the dongle's RTC time or with a '/d' appended to set the dongle's RTC date.

Figure 2

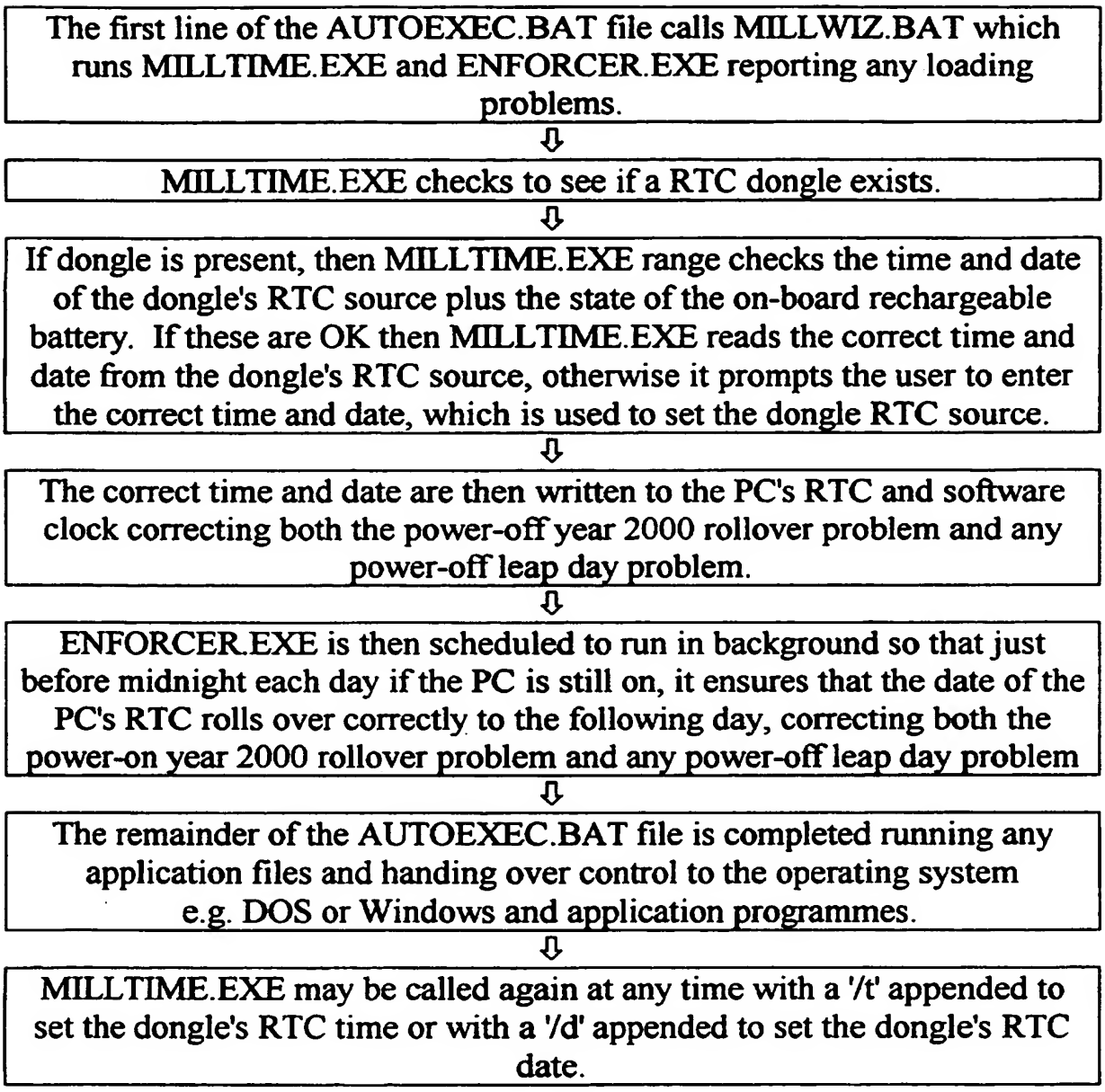
The first line of the AUTOEXEC.BAT file calls MILLWIZ.BAT which runs MILLTIME.EXE and ENFORCER.EXE reporting any loading problems.
MILLTIME.EXE checks to see if a RTC dongle exists.
If dongle is present, then MILLTIME.EXE range checks the time and date of the dongle's RTC source plus the state of the on-board rechargeable battery. If these are OK then MILLTIME.EXE reads the correct time and date from the dongle's RTC source, otherwise it prompts the user to enter the correct time and date, which is used to set the dongle RTC source.
The correct time and date are then written to the PC's RTC and software clock correcting both the power-off year 2000 rollover problem and any power-off leap day problem.
ENFORCER.EXE is then scheduled to run in background so that just before midnight each day if the PC is still on, it ensures that the date of the PC's RTC rolls over correctly to the following day, correcting both the power-on year 2000 rollover problem and any power-off leap day problem
The remainder of the AUTOEXEC.BAT file is completed running any application files and handing over control to the operating system e.g. DOS or Windows and application programmes.
MILLTIME.EXE may be called again at any time with a '/t' appended to set the dongle's RTC time or with a '/d' appended to set the dongle's RTC date.

Figure 3

2323455

A COMPUTER DONGLE

This invention relates to a dongle for a computer, a computer having a such dongle, and a method of recording correct date information in such a computer.

In an attempt to prevent the running of unauthorised copies of computer programs, it is known to supply a computer program with a device known as a "dongle". Such a device comprises a housing having a connector for coupling to an external port (and usually the parallel printer port) of a computer and also having a socket of the same configuration as that of the computer with which the dongle is to connect. Within the dongle there is circuitry, including a programd read-only memory (for instance, an EPROM) suitably connected to the pins of the connectors. When the program with which the dongle is supplied is to be run on the computer, the dongle must first be connected to the appropriate computer port. The program then checks for the presence of the correct dongle, for example by reading a code from the read-only memory and checking that against information in the program, before allowing the program to run properly. An important attribute of a dongle is that it is transparent to the accessory or device which usually would be connected to the same port of the computer, but instead is connected to the corresponding port of the dongle, when that is to be used. As mentioned, a dongle is usually connected to the printer port and thus the printer will instead be connected to the corresponding port of the dongle, so that normal printer commands may be transmitted through the dongle to the printer.

A particular problem being addressed by the computer industry at the present time is that of the so-called "millennium bug", where a computer fails correctly to recognise year 2000 - i.e. the computer is not "year 2000 compliant". This has arisen as a consequence of many computer programs storing date information, and in particular year information, using only the last two digits of the Gregorian year. Many programs either will fail to recognise "00" as meaning the year 2000 or instead might recognise "00" as meaning 1900 since for almost all date processing prior to 2000, programs have simply added 1900 to the last two digits of the year information, in order to arrive at the correct year.

With a so-called personal computer (PC), there are three potential clock problems associated with the millennium bug. There is a hardware real time clock usually built into the mother-board and for post-millennium correct operation, this must be capable of generating date information which may correctly be interpreted by any program addressing the real time clock.

The mother-board also carries in read-only memory a start-up program, referred to as the BIOS, and this includes software which reads the hardware real time clock and sets an incremental counter serving as a clock and which is used by many processes which may be run on the computer and require time information.

Thirdly, there is an operating system clock which is a part of the overall operating system. With the majority of modern PCs, this is either Windows 3.xx or Windows 9x, all of which are products of Microsoft Corporation. The software clocks within these operating systems are capable of correctly

handling date information, provided that they are supplied with the correct information by the hardware real time clock at start-up.

A series of checks have been devised to assess whether a computer is fully year-2000 compliant. These typically check:

1. The ability of the hardware real time clock to increment properly from 31/12/1999 to 01/01/2000, with the computer running over that period;

2. The ability of the hardware real time clock to increment properly from 31/12/1999 to 01/01/2000, with the power off over that period;

3. The ability of the BIOS properly to interpret year information from the hardware real time clock at start-up, including dates to year 2000 and onwards;

4. The ability of the operating system clock to increment properly from 31/12/1999 to 01/01/2000, with the computer running over that period; and

5. The ability of both the hardware real time clock and the operating system clock to recognise that year 2000 is a leap year.

Checks made through a large number of PCs have shown that relatively few manufactured before 1998 have hardware real time clocks able properly to support date information for the year 2000 onwards. In view of the large number of computers which fail at least one, but probably more than just one, of tests 1 to 3, there is a demand for an effective solution which is both easy to implement and significantly cheaper than replacing at least the PC mother-board or even the entire PC. Though many computer operators might seize upon the millennium bug as an opportunity to upgrade hardware at least by

replacing the PC mother-board, if not the entire PC, by one which is fully year 2000 compliant, many other operators will not wish to go to that expense. It is therefore an aim of the present invention to provide a solution to the problem of a PC which is not fully year 2000 compliant, which solution is both easy to implement and relatively inexpensive as compared to the replacement of a mother-board with a compliant real time clock and BIOS.

According to the present invention, there is provided a dongle comprising a housing having a first multiple way connector suitable for connection to an accessory port of a computer and a second multiple way connector suitable for the connection thereto of a computer accessory, there being within the housing a real time clock source, a power supply for running the real time clock source, and control circuitry arranged to respond to a predefined signal appearing at the first connector and to supply to the first connector a time signal derived from the real time clock source of the dongle.

A complete time statement will include year, month, day, hour and second information. However, in this specification, the term "time" can refer just to the year information of a computer time statement.

It will be appreciated that the dongle of this invention may be employed wholly to solve the millennium bug problem of a computer which is not year 2000 compliant. By connecting the dongle to an accessory port such as the printer port of the computer and then automatically running a suitable control program on booting the computer, the real time clock source of the dongle may be read by that program and the information obtained from that real time clock source used to overwrite the computer's hardware real time clock with correct

time (and at least year) information, derived from the dongle real time clock source. Thereafter, any program running on the computer and which accesses the operating system clock or the hardware real time clock will have correct time and date information, derived from the dongle real time clock source.

The real time clock source of the dongle may comprise a clock on a dedicated chip together with the associated circuitry, and a high accuracy clock source may advantageously be employed. In the alternative, the real time clock source may take the form of a microprocessor running a suitable clock program. In this case, the microprocessor may also serve the function of the control circuitry in the dongle, to monitor the lines of the first connector and to supply the requested time and date information when the appropriate signal appears at the first connector.

The power supply for running the dongle real time clock advantageously comprises a rechargeable battery and the control circuitry in the dongle is arranged to effect recharging of the battery by power derived from an operating computer to which the dongle is connected. The rechargeable battery should have sufficient capacity to permit extended running of the dongle real time clock source, in order that time information will not be lost even if the computer to which the dongle is connected is not turned on for an extended period. Typically, the rechargeable battery should be capable of running the clock for several weeks, before recharging becomes essential. In the alternative, the dongle could contain a power source such as a lithium cell, serving to run the real time clock source. To conserve the life of that cell, the circuitry could arrange for the real time clock source to be driven by power derived from a

computer to which the dongle is connected whenever the computer is turned on, the dry cell powering the real time clock source only when the computer is turned off.

Though the dongle could be provided with suitable switches to permit setting of the real time clock source directly on the dongle, it is preferred that the control circuitry permits setting of the real time clock source by signals appearing at the first connector, from an operating computer to which the dongle is connected. For this purpose, a suitable control program must be run on the computer, in order to drive the port to which the dongle is connected with the required signals for clock setting.

This invention extends to a personal computer including a mass storage device having an operating system loaded thereon and an accessory port (such as a parallel printer port) in combination with a dongle of this invention as described above and connected to the accessory port. Preferably, there is a dongle control program loaded on the mass storage device which dongle control program may be run to supply said predefined signal to the port and then to read the real time clock signal supplied by the dongle. Thereafter, the dongle control program may set the operating system software clock and also the hardware real time clock with date and time information obtained from the real time clock source of the dongle.

Advantageously, the boot-up routine of the computer is arranged automatically to run the dongle control program on a power-on boot-up of the computer, or on a re-boot. Having set the hardware and software clocks of the computer, the dongle control program may terminate but may be re-run on

demand of an operator, for example in the event that the date and time information in the dongle real time clock source is to be adjusted.

In an alternative operating mode, the dongle control program loads a TSR program which intercepts all calls to the hardware real time clock in order to validate and, if appropriate adjust, time information responding to those calls. For example, if an application program makes a call to read the hardware real time clock, the TSR program sets the century byte to 19 if the read year is 98 or 99, and sets the century byte to 20 if the read year is less than 98, before passing the date back to the calling application program.

Preferably, the real time clock source of the dongle is a high quality, high performance clock able to maintain time and date information very accurately. Thus, the dongle and control program of this invention may be employed even if a computer is fully year 2000 compliant, but its hardware real time clock either runs relatively inaccurately, or has ceased operating altogether.

According to another aspect of this invention, there is provided a method of setting a software clock of a personal computer having an accessory port, a dongle of this invention as described above connected to the port, a mass storage device, and a bootable operating system and a dongle control program loaded on the mass storage device, in which method on booting the operating system, the dongle control program runs automatically to cause the following steps to be performed:

- said predefined signal is supplied to the port,

- the dongle control circuitry responds to said predefined signal and supplies to the port a time signal derived from the real time clock source of the dongle, and

- the dongle control program reads said time signal from the port and then sets the hardware real time clock and preferably also the software clock to correspond to the time of the dongle real time clock source.

In order that this invention may better be understood, it will now be described in further detail with reference to the accompanying drawings, in which:

Figure 1 is a flow chart for the start-up process of a PC which is not year 2000 compliant;

Figure 2 is a flow chart for the start-up process of a non-compliant PC fitted with a dongle of this invention; and

Figure 3 is a flow chart of an alternative start-up process for a non-compliant PC fitted with a dongle of this invention.

Referring initially to figure 1, there is shown the power-on boot-up sequence for a conventional PC which is not year 2000 compliant. As can be seen, following initial switching-on of the computer, the PC BIOS start-up code is initially run and in the course of this, the computer hardware clock is read by the BIOS program. With a typical non-year 2000 compliant BIOS of a PC, in the event that the last two year digits are less than the two digits of the design year for the BIOS, the BIOS sets the hardware real time clock with a default year, ignoring the century byte. For example, "98" read from the hardware clock is assumed to be 1998 and therefore correct, but "00" is regarded as out

of range and so the BIOS utilises a default year which might be (for example) "94" giving either 1994 or 2094, depending upon the previous state of the century byte, that information then being used to set incorrectly the hardware real time clock.

Following the above clock read, the running of the PC BIOS start-up code is completed, the CONFIG.SYS file is run installing the device drivers and so on, and the AUTOEXEC.BAT file is run in the usual way, thus completing the boot-up procedure. If the actual year is 2000 (or later), a check on the software clock will show that the year has not correctly been recognised, as will a direct call to the PC hardware real time clock.

Referring now to Figure 2, there is shown the flow diagram of a first solution to the above problem, as provided by the present invention. The power-on boot-up procedure of Figure 1 is performed up to the running of the AUTOEXEC.BAT file. The first line of that file calls MILLWIZ.BAT which, in turn, runs MILLTIME.EXE and also ENFORCER.EXE, reporting any problems. MILLTIME.EXE checks to see if the dongle of this invention is fitted to the computer and, if one is, this program range-checks the time and date of the dongle's real time clock source, and also the state of the dongle rechargeable battery. Provided these tests are satisfactorily completed, then MILLTIME.EXE reads the correct time and date information from the dongle's real time clock source. If the tests are not passed, MILLTIME.EXE prompts the user manually to enter the correct time and date information, which is used to set the dongle real time clock source. Following that, the new and correct time and date are written to the PC's clocks, overwriting the information incorrectly set during the

power-on boot-up procedure, and thus fixing both the power-off year 2000 roll-over problem and any power-off leap day problem.

ENFORCER.EXE then loads a memory resident routine which is arranged to intercept any read or write access to the PC's hardware real time clock, and the remainder of the AUTOEXEC.BAT file is run, eventually handing over control to the operating system such as Windows 3.*xx* or Windows 9*x*.

Provided that the PC remains switched on, the memory resident routine loaded by ENFORCER.EXE intercepts all calls for information from the PC's hardware real time clock. The century byte is set to 19 if the read yields a year of 98 or 99, or set to 20 for any value read of less than 98, before the date is returned to the original calling program. In this way, the power-on year 2000 problem and any power-on leap day problem can be resolved, as the current century byte will have been set at start-up.

If the real time clock of the dongle is to be set, this may be done by calling MILLTIME.EXE again, but with an appropriate switch set to permit setting of the dongle's time, or the dongle's date. This may be achieved using conventional on-screen techniques.

Figure 3 shows an alternative solution, still using the dongle of this invention. The procedure of figure 3 is essentially the same as that of figure 2, but differs in the way in which ENFORCER.EXE operates. Here, ENFORCER.EXE is not loaded as a TSR program, but rather is schedules to run in the background, just before midnight each day. ENFORCER.EXE checks the date on the PC's hardware real time clock, to ensure it rolls over properly to the next following day, and if appropriate takes the required action

to remedy any problem, reading the correct time and date information from the dongle's real time clock source.

CLAIMS

1. A dongle comprising a housing having a first multiple way connector suitable for connection to an accessory port of a computer and a second multiple way connector suitable for the connection thereto of a computer accessory, there being within the housing real time source, a power supply for running the real time source, and control circuitry arranged to respond to a predefined signal appearing at the first connector and to supply to the first connector a time signal derived from the real time clock source of the dongle.

2. A dongle as claimed in claim 1, wherein the real time clock source comprises an integrated clock component configured to yield real time information.

3. A dongle as claimed in claim 1, wherein the real time clock source comprises a microprocessor running a clock program.

4. A dongle as claimed in claim 3, wherein the microprocessor further provides the control circuitry monitoring the first connector for the predefined signal, and drives the first connector with time information.

5. A dongle as claimed in any of the preceding claims, wherein the control circuitry powers the real time clock source from power derived from the computer through the first connector whenever the dongle is connected to an operating computer.

6. A dongle as claimed in any of the preceding claims, wherein the power source within the housing for the real time clock source comprises a

rechargeable battery and the control circuitry effects recharging of the battery from an operating computer to which the dongle is connected.

7. A dongle as claimed in any of the preceding claims, wherein the control circuitry permits setting of the real time clock source by signals appearing at the first connector from an operating computer to which the dongle is connected.

8. A dongle as claimed in any of the preceding claims, wherein the first and second connectors both comprise conventional PC parallel port connectors but of opposite polarity.

9. A dongle as claimed in claim 1 and substantially as hereinbefore described with reference to the accompanying drawings.

10. A personal computer including a mass storage device having an operating system loaded thereon and an accessory port, in combination with a dongle as claimed in any of the preceding claims and connected to the accessory port.

11. The combination of claim 10, wherein there is a dongle control program loaded on the mass storage device which dongle control program is arranged to supply said predefined signal to the port and then to read the real time clock signal supplied by the dongle.

12. The combination of claim 11, wherein the dongle control program is arranged to set the hardware real time clock of the computer with time information obtained from the real time clock source of the dongle.

13. The combination of claim 11 or claim 12, wherein the dongle control program is arranged to set the operating system software clock with time information obtained from the real time clock source of the dongle.

14. The combination of any of claims 11 to 13, wherein the dongle control program is arranged to run automatically on booting the operating system from the mass storage device.

15. A method of setting a software clock of a personal computer having an accessory port, a dongle as claimed in any of claims 1 to 9 connected to the port, a mass storage device, a bootable operating system and a dongle control program loaded on the mass storage device, in which method on booting the operating system, the dongle control program runs automatically to cause the following steps to be performed:

- said predefined signal is supplied to the port,

- the dongle control circuitry responds to said predefined signal and supplies to the port a time signal derived from the real time clock source of the dongle, and

- the dongle control program reads said time signal from the port and then sets the hardware real time clock to correspond to the time of the dongle real time clock source.

16. A method as claimed in claim 15, in which the dongle control program sets the operating system clock of the computer with time information corresponding to the time read from the dongle real time clock source.

17. A method as claimed in claim 15 or claim 16, in which said time signal carries time and date information for setting the computer's clocks.

18. A method as claimed in any of claims 15 to 17, wherein the dongle is transparent to signals intended to be passed to or from an accessory connected through the dongle to the port.

19. A method as claimed in any of claims 15 to 18 and substantially as hereinbefore described.

The Patent Office

16

Application No: GB 9811313.7 **Examiner:** Ross Cavill
Claims searched: 1 to 19 **Date of search:** 26 June 1998

Patents Act 1977
Search Report under Section 17

Databases searched:

UK Patent Office collections, including GB, EP, WO & US patent specifications, in:

UK Cl (Ed.P): G3T (TAA-F,THA,THB)

Int Cl (Ed.6): G04G 1/00; G06F1/00,/14

Other: Online:WPI

Documents considered to be relevant:

Category	Identity of document and relevant passage	Relevant to claims
A	US 4706189 (BROCKMAN) whole document	All

X	Document indicating lack of novelty or inventive step	A	Document indicating technological background and/or state of the art.
Y	Document indicating lack of inventive step if combined with one or more other documents of same category.	P	Document published on or after the declared priority date but before the filing date of this invention.
&	Member of the same patent family	E	Patent document published on or after, but with priority date earlier than, the filing date of this application.